

## CLAIMS

What is claimed is:

- 1     1.     A method for managing an instance of a model, wherein the instance includes a set of  
2           nodes with associated node variables and a set of connection constraints, the method  
3           comprising the computer-implemented steps of:  
4           receiving an intent to modify the instance, wherein the intent to modify specifies a  
5           first node of the set of nodes;  
6           loading into volatile memory information associated with said first node;  
7           determining, based at least on said information associated with said first node, a first  
8           subset of said set of connection constraints, wherein said first subset of  
9           connection constraints includes all connection constraints that restrict said  
10          intent to modify, and wherein said first subset of connection constraints  
11          includes fewer constraints than said set of connection constraints;  
12          determining, based on said first subset of connection constraints, a first subset of node  
13          variables that are associated with nodes of said set of nodes, wherein said first  
14          subset of node variables includes all variables that may have associated values  
15          that affect whether any constraint of said first subset of connection constraints  
16          is violated; and  
17          loading into volatile memory a first subset of node variable information, wherein said  
18          first subset of node variable information includes only information about said  
19          first subset of node variables.
- 1     2.     The method of Claim 1, further comprising the computer-implemented steps of:  
2           receiving a modification to said first node; and

3 based on said first subset of node variable information and said set of connection  
4 constraints, determining whether said modification violates any constraint of  
5 said set of connection constraints.

1 3. The method of Claim 2, further comprising the computer-implemented steps of:  
2 upon a determination that a variable value associated with a second node of said set  
3 of nodes is affected by said modification,  
4 determining, based on said first subset of node variables, a second subset of  
5 said set of connection constraints, wherein said second subset of  
6 connection constraints includes all constraints that restrict said variable  
7 value associated with said second node;  
8 determining, based on said second subset of connection constraints, a second  
9 subset of node variables that are associated with nodes of said set of  
10 nodes, wherein said second subset of node variables includes all  
11 variables that may have associated values that affect whether any  
12 constraint of said second subset of connection constraints is violated;  
13 and  
14 to determine whether said variable value of said second node violates any  
15 constraints of said set of connection constraints, loading into volatile  
16 memory, if not already loaded, a second subset of node variable  
17 information, wherein said second subset of node variable information  
18 includes only information about said second subset of node variables.

1 4. The method of Claim 1, wherein said first subset of connection constraints includes  
2 only constraints that restrict said intent to modify.

- 1 5. The method of Claim 1, further comprising the computer-implemented step of  
2 indicating that a constraint of said set of connection constraints is violated by said  
3 intended modification.
- 1 6. The method of Claim 1, wherein said intended modification includes a change to said  
2 first node.
- 1 7. The method of Claim 6, further comprising the computer-implemented step of  
2 indicating that a constraint of said set of connection constraints is violated by said  
3 change to said first node.
- 1 8. The method of Claim 1, wherein said modification includes addition of said first node  
2 to said instance.
- 1 9. The method of Claim 8, further comprising the computer-implemented step of  
2 providing a set of one or more variable values for said first node that do not violate  
3 any of said constraints of said set of connection constraints.
- 4 10. The method of Claim 1, further comprising the computer-implemented step of  
5 causing display of a representation of said first subset of connection constraints and  
6 said first subset of node variables.
- 1 11. The method of Claim 1, wherein said first subset of node variables includes variables  
2 associated with a first set of one or more passive nodes, and wherein said first node is  
3 editable and said one or more passive nodes are not editable.

- 1 12. The method of Claim 1, further comprising the computer-implemented step of  
2 representing said model according to constraint programming techniques.
- 1 13. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 1.
- 1 14. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 2.
- 1 15. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 3.
- 1 16. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 4.
- 1 17. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 5.
- 1 18. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 6.

1 19. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 7.

1 20. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 8.

1 21. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 9.

1 22. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 10.

1 23. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 11.

1 24. A computer-readable medium carrying one or more sequences of instructions which,  
2 when executed by one or more processors, causes the one or more processors to  
3 perform the method recited in Claim 12.

1 25. A system for managing an instance of a model that includes a set of nodes and a set of  
2 constraints, the system comprising:

3 means for receiving an intent to modify the instance, wherein the intent to modify  
4 specifies a first node of the set of nodes;  
5 means for loading into volatile memory information associated with said first node;  
6 means for determining, based at least on said information associated with said first  
7 node, a first subset of said set of connection constraints, wherein said first  
8 subset of connection constraints includes all connection constraints that  
9 restrict said intent to modify, and wherein said first subset of connection  
10 constraints includes fewer constraints than said set of connection constraints;  
11 means for determining, based on said first subset of connection constraints, a first  
12 subset of node variables that are associated with nodes of said set of nodes,  
13 wherein said first subset of node variables includes all variables that may have  
14 associated values that affect whether any constraint of said first subset of  
15 connection constraints is violated; and  
16 means for loading into volatile memory a first subset of node variable information,  
17 wherein said first subset of node variable information includes only  
18 information about said first subset of node variables.

1 26. The system of Claim 25, further comprising:  
2 means for receiving a modification to said first node; and  
3 means for determining, based on said first subset of node variable information and  
4 said set of connection constraints, whether said modification violates any  
5 constraint of said set of connection constraints.

1    27.    The system of Claim 26, further comprising:  
2           means for determining, upon a determination that a variable value associated with a  
3           second node of said set of nodes is affected by said modification and based on  
4           said first subset of node variables, a second subset of said set of connection  
5           constraints, wherein said second subset of connection constraints includes all  
6           constraints that restrict said variable value associated with said second node;  
7           means for determining, upon a determination that a variable value associated with a  
8           second node of said set of nodes is affected by said modification and based on  
9           said second subset of connection constraints, a second subset of node  
10          variables that are associated with nodes of said set of nodes, wherein said  
11          second subset of node variables includes all variables that may have  
12          associated values that affect whether any constraint of said second subset of  
13          connection constraints is violated; and  
14          means for loading into volatile memory, if not already loaded, a second subset of  
15          node variable information to determine whether said variable value of said  
16          second node violates any constraints of said set of connection constraints,  
17          wherein said second subset of node variable information includes only  
18          information about said second subset of node variables.